

SafeCircle Landing Page - Developer Handoff

Overview

Bilingual (English/Dutch) landing page for SafeCircle wellness monitoring app. Built with React, TypeScript, Vite, Tailwind CSS v4, and i18next for translations.

Data Collection Goals

Track the complete checkout funnel for demand validation:

- Website visits (all users, anonymous)
- Plan selection clicks (Basic vs Full plan)
- Email submissions (checkout form)
- Payment completions (Stripe success)

Privacy-first approach: Anonymous server-side tracking for ALL users + enhanced analytics (Google Analytics, Hotjar) only when users accept analytical cookies.

Key Setup Tasks

1. Stripe Payment Integration

File: [/src/app/pages/Checkout.tsx](#)

What to do:

- Replace `YOUR_STRIPE_PUBLISHABLE_KEY` (line ~10) with your Stripe publishable key
- Create products in Stripe Dashboard:
 - Basic Plan: €5/month (price ID needed)
 - Full Plan: €8/month (price ID needed)
- Update price IDs in Checkout.tsx
- Set up Stripe webhook endpoint to handle `checkout.session.completed` events
- Test with Stripe test mode first

Docs: See [/INTEGRATION_SETUP.md](#) for detailed Stripe setup

2. Google Analytics & Hotjar

Files:

- [/src/app/App.tsx](#) (lines 50-55)
- [/src/utils/tracking.ts](#)
- [/src/utils/initAnalytics.ts](#)

What to do:

- Replace `G-XXXXXXXXXX` in [App.tsx](#) line 51 with your GA4 Measurement ID
- Uncomment Hotjar initialization (line 55) and add your Hotjar ID
- Analytics only initialize when users accept analytical cookies (GDPR compliant)

Docs: See [/TRACKING_SETUP.md](#) and [/TEST_TRACKING.md](#)

3. Contact Form Email Setup

File: [/supabase/functions/send-contact-email/index.ts](#)

What to do:

- Connect to Supabase (or any backend)
- Configure SMTP service (SendGrid, AWS SES, etc.)
- Update recipient email address in the edge function
- Set environment variables for SMTP credentials

Docs: See [/supabase/SETUP_INSTRUCTIONS.md](#)

4. Language-Specific SEO (Dutch vs English)

Current setup:

- Main route `/` = `LiveAlonePage` (English by default, switches based on i18n)
- All pages support both languages via language switcher

For proper SEO:

Option A - Separate URL paths (Recommended):

```
// Update routes in /src/app/App.tsx
<Route path="/" element={<LiveAlonePage />} />           // English
<Route path="/nl" element={<LiveAlonePage />} />         // Dutch
<Route path="/worry-about-someone" element={...} />        // English
<Route path="/nl/bezorgd-over-iemand" element={...} />    // Dutch
```

Then add to each page component:

```
// In LiveAlonePage component
useEffect(() => {
  if (location.pathname.startsWith('/nl')) {
    i18n.changeLanguage('nl');
  } else {
    i18n.changeLanguage('en');
  }
}, [location.pathname]);
```

Option B - Use hreflang tags: Add to [MetaTags.tsx](#) component:

```
<link rel="alternate" hreflang="en" href="https://yoursite.com/" />
<link rel="alternate" hreflang="nl" href="https://yoursite.com/nl" />
<link rel="alternate" hreflang="x-default" href="https://yoursite.com/" />
```

Google Search Console:

- Submit both language versions to Google Search Console
- Create separate sitemaps for [/en/](#) and [/nl/](#) URLs
- Use hreflang annotations

Translation System

All copy lives in:

- [/src/i18n/locales/en.json](#) (English)
- [/src/i18n/locales/nl.json](#) (Dutch)
- CSV templates: [/safecircle-complete-translation.csv](#)

Important Files

- [/TRACKING_EVENTS.md](#) - All tracking events documented
- [/TRACKING_QUICK_START.md](#) - Quick setup guide
- [/INTEGRATION_SETUP.md](#) - Stripe + analytics integration
- [/TEST_TRACKING.md](#) - How to test tracking

Environment Variables Needed

```
VITE_STRIPE_PUBLISHABLE_KEY=pk_test_...
VITE_GA_MEASUREMENT_ID=G-XXXXXXXXXX
VITE_HOTJAR_ID=your_hotjar_id
```

Testing Before Launch

1. Test both language versions
2. Test Stripe checkout flow (test mode)
3. Verify tracking events fire (check browser console)
4. Test cookie consent banner functionality
5. Test contact form submission
6. Check mobile responsiveness (hamburger menu, hero image hidden on mobile)